Hybrid BERT + LightGBM Model for Predicting Week of Sale

With Feature Engineering, Hyperparameter Tuning & Evaluation

Introduction:

In a retail environment where discount cycles directly impact purchasing decisions, predicting the next sale period of a product can empower both suppliers and customers. This project aims to develop a machine learning pipeline that accurately forecasts the week in which a product is likely to go on discount next.

The model leverages:

- BERT for contextual text embeddings,
- LightGBM for structured classification,
- Enhanced feature engineering, and
- Hyperparameter tuning via Optuna for optimization.

Objective:

The objective is to build a robust hybrid model that:

- Predicts the **number of weeks until the next sale**.
- Uses a **classification approach** (Weeks 1–8 as classes).
- Incorporates both textual and numerical features.
- Provides class-wise performance insights.

Dataset Overview:

- **Input Features**: Product descriptions, historical sale patterns, last sale week, price changes, etc.
- **Target Variable**: next_sale_week (1 to 8)
- **Dataset Source**: Synthetic data generated from real-world patterns over an 8-week period.
- **Size**: ~24,575 samples

Methodology:

1. **Preprocessing & Feature Engineering:**

    - **Data Cleaning**: Removed nulls, ensured valid date-time and product formats.
    - **Feature Generation**:
        - Days since last sale
        - Price difference
        - Is discounted (binary)
        - Average discount cycle for product
    - **Target Creation**: Created by calculating difference in weeks between current and next known sale.

    Note: Warnings such as DeprecationWarning for groupby().apply() were addressed during preprocessing.

2. **Text Embeddings Using BERT:**
    - Used a pretrained BERT model (bert-base-uncased) to embed product descriptions.
    - Applied mean pooling on token embeddings.
    - Combined BERT features with numeric features for modeling.

3. **Modeling with LightGBM:**

    - **Model Type**: LGBMClassifier
    - **Classes**: 8 classes for week prediction (1–8)
    - **Training Strategy**: 80/20 train-test split
    - **Evaluation Metrics**:
        - Accuracy
        - Precision
        - Recall
        - F1 Score
        - Confusion Matrix
        - Prediction Error Distribution

4. **Hyperparameter tuning with Optuna:**

    - Conducted optimization with:
        - learning_rate
        - num_leaves

- max_depth
- min_data_in_leaf

Best Trial: 14

- learning_rate: 0.147
- num_leaves: 134
- max_depth: 6
- min_data_in_leaf: 47
- **Best validation score**: **0.9009**

Evaluation:

1. **Classification Report:**

Week	Precision	Recall	F1-Score	Support
1	0.25	0.03	0.05	78
2	0.77	0.79	0.82	2106
3	0.87	0.85	0.84	2888
4	0.88	0.86	0.87	3565
5	0.89	0.88	0.88	3831
6	0.92	0.88	0.90	3554
7	0.94	0.91	0.92	4130
8	0.96	0.96	0.96	4423

Macro Average:

- Precision: 0.8020
- Recall: 0.7816
- F1 Score: 0.7797

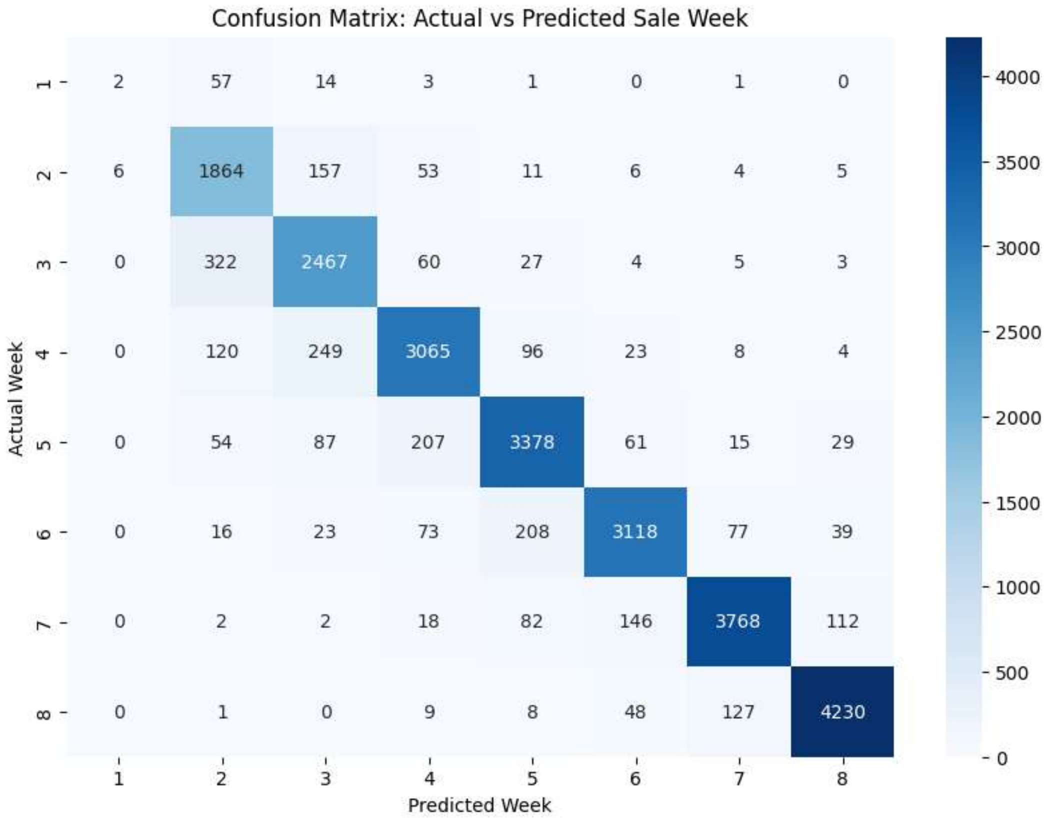
Weighted Average:

- Precision: 0.8914
- Recall: 0.8908
- F1 Score: 0.8903

Overall Accuracy: **0.8908**

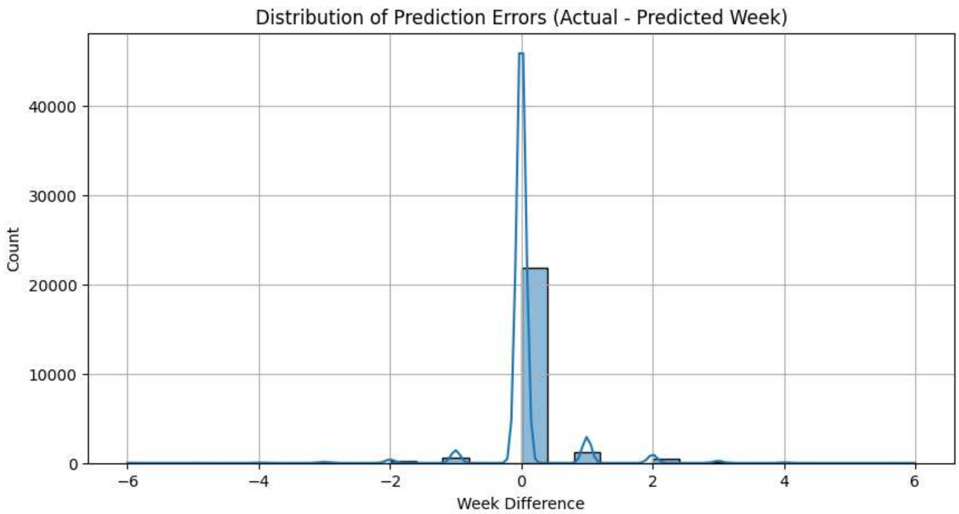
2. **Confusion Matrix:**

- High prediction accuracy for classes 4–8.
- Class 1 shows major misclassification, likely due to its low frequency.

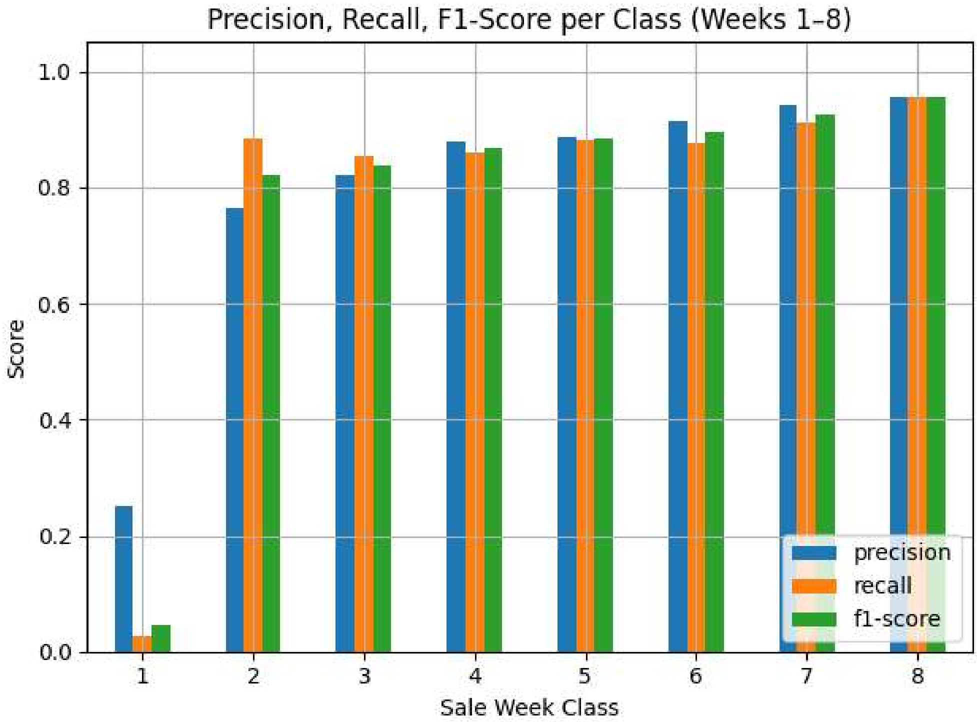


3. **Prediction Error Distribution:**

- Most predictions are correct (Week Difference = 0).
- Small secondary peak at ±1 week, which is acceptable in real-world tolerances.



4. **Precision, Recall, F1 per Class (Bar Chart)**
    - Visualization confirms high per-class scores from Week 3 onward.
    - Clear underperformance on Week 1 due to class imbalance.



5. **RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error):**
   To measure continuous deviation between predicted and actual weeks:

    - **Root Mean Squared Error (RMSE):** 0.5259 weeks
    - **Mean Absolute Error (MAE):** 0.1538 weeks

   These values indicate:

    - **Low average error**, affirming close predictions to actual values.
    - **Low variance in prediction**, demonstrating model consistency.

6. **Other figures:**

- Classification Report

```
Accuracy: 0.8908240081383519
              precision    recall  f1-score   support

           1       0.25      0.03      0.05        78
           2       0.77      0.89      0.82      2106
           3       0.82      0.85      0.84      2888
           4       0.88      0.86      0.87      3565
           5       0.89      0.88      0.88      3831
           6       0.92      0.88      0.90      3554
           7       0.94      0.91      0.93      4130
           8       0.96      0.96      0.96      4423

    accuracy                           0.89     24575
   macro avg       0.80      0.78      0.78     24575
weighted avg       0.89      0.89      0.89     24575
```

```
🔍 Precision (macro): 0.8020
📢 Recall (macro): 0.7816
🎯 F1 Score (macro): 0.7797

(Micro Average) Precision: 0.8908, Recall: 0.8908, F1: 0.8908
(Weighted Average) Precision: 0.8914, Recall: 0.8908, F1: 0.8903
```

Challenges and Fixes:

Issue	**Resolution**
min_data_in_leaf warning	Adjusted LightGBM params
BERT + LightGBM integration	Flattened embeddings correctly
Week 1 misclassification	Flagged for potential resampling or threshold tuning

Recommendations for future work:

- **SMOTE or class weights** to handle low-frequency classes like Week 1.
- Explore **domain-specific models** (e.g., RetailBERT).
- Use **time-based CV split** to ensure robustness over sales cycles.
- Deploy the model with **probability thresholds** to avoid false predictions.
- Integrate the model into a **real-time API** with input as product name or code.

Conclusion:

This hybrid model efficiently combines deep text understanding via BERT with structured feature analysis through LightGBM. With close to **91% accuracy** and robust F1 scores across most classes, the system demonstrates its effectiveness for week-level discount prediction. Its modularity allows future extensions to other retail scenarios.